

**Amendments To The Claims**

The listing of claims will replace all prior versions and listings of claims in the application. The listing of claims presents each claim with its respective status shown in parentheses.

By this Amendment, Appellant is cancelling Claim 38 without prejudice or disclaimer. Claims 5, 6, 8–10, 20, 21, 23–25, 33, 34, 36, 37, and 41 remain as originally filed, and Claims 1–4, 7, 11, 19, 22, 30, 32, 35, 40 and 49 remain as previously presented. Claims 12–18, 26–29, 31, 39 and 42–48 were previously cancelled.

1. (Previously Presented) A method of reorganizing a table in a database file while providing clients of the database file continuous access to data stored in the table, the method comprising:

reorganizing data of an original table by copying the data to a reorganized table;

during the copying, allowing modifications to the data of the original table while collecting records of the modifications;

when the copying completes, applying the modifications from the collected records against the reorganized table;

applying a first partial lock to the original table, the first partial lock blocking select data modification operations against the original table while allowing other operations against the original table;

applying any remaining modifications from the collected records against the reorganized table;

applying a second partial lock to the reorganized table, the second partial lock blocking select data modification operations against the reorganized table while allowing other operations against the reorganized table during the reorganization such that the reorganized table remains continuously accessible during the reorganization;

substituting the reorganized table for the original table such that the reorganized table remains accessible while substituting the reorganized table for the original table;  
and

removing the second partial lock, wherein additional more-restrictive locks to the original table are not needed during the method of reorganizing the original table, thereby providing clients of the original table continuous access to the data during the reorganization through at least the other operations allowed by the first partial lock.

2. (Previously Presented) The method of Claim 1, wherein the other operations allowed by at least one of the first and second partial locks comprises one or more structural modification operations.

3. (Previously Presented) The method of Claim 1, wherein the other operations allowed by at least one of the first and second partial locks comprises one or more read-only operations.

4. (Previously Presented) The method of Claim 1, further comprising:

during the application of the modifications from the collected records against the reorganized table, allowing additional modifications to the data of the original table while collecting additional records of the additional modifications; and

when the modifications and at least portions of the additional modifications have been applied against the reorganized table, applying the first partial lock to the original table;

wherein the step of applying any remaining modifications includes applying any remaining modifications or additional modifications against the reorganized table.

5. (Original) The method of Claim 1, wherein when the original table included one or more relational constraints, the method further comprises applying at least one of the one or more relational constraints to the reorganized table.

6. (Original) The method of Claim 5, wherein the application of the at least one relational constraint to the reorganized table includes applying a trigger procedure to the reorganized table.

7. (Previously Presented) The method of Claim 5, wherein the application of the at least one relational constraint to the reorganized table includes applying a partial lock to another table.

8. (Original) The method of Claim 1, wherein the original table includes a table name, and wherein the step of substituting the reorganized table for the original table further

comprises renaming the original table another name and naming the reorganized table the table name.

9. (Original) The method of Claim 1, further comprising archiving the original table.

10. (Original) The method of Claim 1, wherein the copying of the data of the original table to the reorganized table further comprises creating an original synchronization point, after which the records of modifications are collected.

11. (Previously Presented) The method of Claim 1, wherein before the application of the second partial lock, the original table and the reorganized table are in synchronization with one another.

12. – 18. (Cancelled).

19. (Previously Presented) A method of reorganizing an object in a database, the method comprising:

reorganizing an original object by copying data from the original object to a reorganized object;

applying a first partial lock to the original object, the first partial lock blocking data modification operations from modifying the original object while allowing other operations to access the original object, wherein additional more-restrictive locks to the original object are not needed during the method of reorganizing the original object, thereby providing clients of the original object continuous access to the data during the reorganization through at least the other operations allowed by the partial lock;

applying a second partial lock to the reorganized table, the second partial lock blocking select data modification operations against the reorganized table while allowing other operations against the reorganized table during the reorganization; and

substituting the reorganized object with the original object such that the reorganized object remains readable while substituting the reorganized object for the original object.

20. (Original) The method of Claim 19, wherein the other operations include one or more read-only operations.

21. (Original) The method of Claim 19, wherein the other operations include one or more structural modification operations.

22. (Previously Presented) A computer-implemented method of reorganizing an object in a database file, the method comprising:

reorganizing an original object by copying data from the original object to a reorganized object;

applying a first partial lock to the original table, the first partial lock blocking select data modification operations against the original table while allowing other operations against the original table;

applying a second partial lock to the reorganized object, the second partial lock blocking data modification operations from modifying the reorganized object, while allowing other operations to access the reorganized object, wherein the reorganized object remains accessible during reorganization; and

substituting the reorganized object with the original object such that the reorganized object remains readable while substituting the reorganized object for the original object.

23. (Original) The method of Claim 22, wherein the other operations include one or more read-only operations.

24. (Original) The method of Claim 22, wherein the other operations include one or more structural modification operations.

25. (Original) The method of Claim 22, wherein the one or more structural modification operations include consecutive data definition language operations.

26. – 29. (Cancelled).

30. (Previously Presented) A reorganization system, comprising:

at least one database file having a table of data and a log file;

a database management system communicating with the at least one database file, thereby governing the modification of the data in the table; and

a reorganization application communicating with the database management system to access the table and communicating with the database file to access the log file, wherein the reorganization application is configured to copy the data of the table to a

reorganized table, to apply modifications from the log file corresponding to modifications to the table during the copying of the data, and to substitute the reorganized table for the table, thereby reorganizing the data of the table, wherein the reorganization application is further configured to apply a first partial lock to the table, thereby blocking select data modification language operations while allowing at least read-only operations, wherein additional more-restrictive locks to the table are not needed during reorganization of the table, thereby providing clients of the table access to the data during the reorganization through at least the other operations allowed by the first partial lock, and wherein the reorganization application is further configured to apply a second partial lock to the reorganized table, thereby blocking select data modification language operations while allowing at least read-only operations and wherein the reorganized table remains readable when the reorganization application substitutes the reorganized table for the table.

31. (Cancelled).

32. (Previously Presented) A reorganization application for reorganizing an object in a database, the reorganization application comprising an execution thread which reorganizes an original object by copying data of the original object to a reorganized object, and which applies a first partial lock to the original object, wherein the first partial lock blocks data modification operations from modifying the original object while allowing other operations to access the original object thereby providing clients of the original object access to the data during the reorganization through at least the other operations allowed by the first partial lock, and wherein a second partial lock is applied to the reorganized object such that the reorganized object remains readable when substituting the reorganized object for the original object.

33. (Original) The reorganization application of Claim 32, wherein the other operations include one or more read-only operations.

34. (Original) The reorganization application of Claim 32, wherein the other operations include one or more structural modification operations.

35. (Previously Presented) A computer-implemented reorganization application for reorganizing an object in a database, the reorganization application comprising an execution thread which reorganizes an original object by copying data of the original object to a reorganized object, and which applies a first partial lock to the original object and a second

partial lock to the reorganized object, wherein the second partial lock blocks data modification operations from modifying the reorganized object while allowing other operations to access the reorganized object, and wherein the reorganized object remains readable when substituting the reorganized object for the original object.

36. (Original) The reorganization application of Claim 35, wherein the other operations include one or more read-only operations.

37. (Original) The reorganization application of Claim 35, wherein the other operations include one or more structural modification operations.

38. (Cancelled).

39. (Cancelled).

40. (Previously Presented) A computer-implemented reorganization application for reorganizing an object in a database, the reorganization application comprising an execution thread which reorganizes an original object by copying data of the original object to a reorganized object and which substitutes the reorganized object for the original object, wherein the execution thread applies a first partial lock to the original object and a second partial lock to the reorganized object that allows read-only access to the data during the substitution of the reorganized object for the original object while blocking other access to the data and wherein the reorganized object remains continuously accessible during reorganization.

41. (Original) The reorganization application of Claim 40, wherein the read-only access to the data includes read-only access during multiple data definition language operations.

42. – 48 (Cancelled).

49. (Previously Presented) The reorganization system of Claim 30, wherein the second partial lock blocks select data modification language operations while allowing one of one or more read-only operations and one or more data definition language operations.